An Architecture for Wireless Simulation in NS-2 Applied to Impulse-Radio Ultra-Wide Band Networks[*] [†]

Ruben Merz
EPFL
School of Computer and Communication Sciences
ruben.merz@epfl.ch

Jean-Yves Le Boudec
EPFL
School of Computer and Communication Sciences
jean-yves.leboudec@epfl.ch

Joerg Widmer[‡]
DoCoMo Euro-Labs
lastname@docomolab-euro.com

Keywords
UWB, ns-2, wireless networking, multi-user inteference

ABSTRACT
We present an architecture for implementing a wireless physical layer in a packet-based network simulator. We integrate this architecture in the popular ns-2 network simulator and use it to implement an impulse-radio ultra-wide band (IR-UWB) physical layer. Contrary to the current wireless physical layer implementation of ns-2, in our case a packet is fully received by our physical layer before being delivered to the MAC layer. A packet detection and timing acquisition model has been implemented. Furthermore, for each packet, a packet error rate (PER) can be computed as a function of the received power, interference from concurrent transmissions, and thermal noise. This architecture is quite generic and allows for the simulation of any physical layer where an accurate model of interference is of high importance, e.g., IR-UWB or CDMA. Our implementation for IR-UWB takes into account transmissions with different time-hopping sequences (THS). The underlying modulation is binary phase shift keying (BPSK), followed by a variable-rate channel code. Our implementation is the first available that allows for the simulation of IR-UWB networks. It is modular and can thus be easily modified and extended.

[*]The work presented in this paper was supported (in part) by the National Competence Center in Research on Mobile Information and Communication Systems (NCCR-MICS), a center supported by the Swiss National Science Foundation under grant number 5005-67322, and by CTI contract No7109.2;1 ESPP-ES

[†]Ruben Merz and Jean-Yves Le Boudec are with EPFL, School of Computer and Communication Sciences, CH-1015 Lausanne, Switzerland. Jörg Widmer is with DoCoMo Euro-Labs, Landsberger Strasse 312, 80687 Munich, Germany.

[‡]The work of this author was conducted while he was at EPFL.

1. INTRODUCTION
The network simulator ns-2 [1] is a popular and widely used discrete-event based simulator for wireless and wired network simulation[1]. NS-2 allows researchers to simulate a wide range of network configurations. It supports various protocols at the application layer, and mainly TCP and UDP at the transport layer. It also includes models for simulation of wired and wireless physical layers. Regarding its wireless physical layer implementation, ns-2 mainly offers an implementation for the IEEE 802.11 protocol[2].

Developments and research in the fields of wireless communication and networking today prompts tools to evaluate and understand the performance of new protocols and new wireless physical layers. Simulation tools such as ns-2 are obviously very important in this aspect. Indeed, when evaluating the performance of wireless protocols on a complex topology (for instance, 802.11 in a multi-hop scenario), simulation is an invaluable and necessary tool.

Unfortunately, with the current implementation of the popular ns-2 network simulator, it is far from easy to implement new wireless physical layers or to modify the existing 802.11 implementation. There are several reasons (see Section 2), but probably the most important is a large dependence of the current wireless physical layer of ns-2 on the IEEE 802.11 physical layer.

In our case, our interest lies in impulse-radio ultra-wide band (IR-UWB) radio networks [8]. A typical IR-UWB physical layer [26, 23] makes use of ultra-short duration (< 1 ns) pulses that yield ultra-wide bandwidth signals. They are characterized by a low duty cycle ($\simeq 1\%$) and extremely low power spectral densities [26]. Multi-user access is possible thanks to pseudo-random *time hopping* sequences (THS) that randomize the transmit time of each pulse. This technique is called time-hopping.

Our objective for the present work is to be able to simulate a medium access control (MAC) protocol for IR-UWB

[1]The latest release at this time of writing is ns-2.30, see `http://nsnam.isi.edu/nsnam/index.php/Main_Page`

[2]There is also an implementation of the IEEE 802.15.4 protocol, but its support and user base is not as large as 802.11

radio networks (the DCC-MAC protocol, see [20]). In addition, with the development of the IEEE 802.15.4a[3], there is clearly a need for the support and availability of an IR-UWB physical layer in a simulator such as ns-2.

Our contributions are the following:

- A modified architecture of the wireless physical layer in ns-2. Support for multiple sub-channels (see Section 2.1.2), support for bit error rate (BER) and packet error rate (PER) computation, support for different modulations (through the addition of a modulation object), support for cumulative interference, and support for packet detection and acquisition.

  Our modified architecture of the wireless physical layer allows for an easy integration of other physical layer models in ns-2.

- A particular implementation of an IR-UWB physical layer (see Section 2.3) based on our modified architecture in ns-2. The physical layer modeled by this implementation assumes a binary phase-shift keying (BPSK) modulation with a variable rate convolutional code. For each received packet, a signal to interference and noise ratio (SINR) is calculated. The bit error rate (BER) corresponding to this SINR is then obtained by lookup tables computed offline[4]. For simplicity reasons, a Gaussian approximation is currently used for the multi-user interference and for computing the SINR. From the computed BER, the PER is obtained by standard approximation. The PER is used as the parameter of a binary random variable used to decide whether the packet is properly received or not. We also implement a propagation model for UWB channels [10]. All our code is freely and publicly available [3].

It is well-known that the Gaussian approximation is not sufficiently accurate [6, 15, 11, 18]. But it is still a viable solution in the short term. It allows for focusing the development and implementation on architectural issues and for debugging. In the long term, a more accurate model for the computation of the BER and PER is necessary. We are currently investigating these issues. For instance, see [18] for a possible solution that incorporates a multipath propagation channel, makes no assumption about the power levels at the receiver and accurately takes into account the multi-user interference.

Compared to narrowband radio networks such as 802.11, IR-UWB networks exhibit the following property: thanks to time-hopping, concurrent transmissions of packets do not necessarily result in destructive interference such as collisions in 802.11. Hence, it is possible to have concurrent transmissions without the need for an exclusion protocol[8]. However, packet detection and timing synchronization is more difficult to achieve than in narrowband radio networks and can fail with a non-negligible probability.

The remainder of this paper is organized as follows: in Section 2.1, we present the main issues of the current wireless physical layer implementation in ns-2. In Section 2.2, we present the difference of our implementation with respect to the one in ns-2 and in Section 2.3, we focus on the details specific to our IR-UWB implementation. In Section 3, we illustrate the path of a packet trough our physical layer implementation and present simulation results in Section 4. We discuss the related work in Section 5 and conclude this paper in Section 6.

2. AN ARCHITECTURE FOR WIRELESS PHYSICAL LAYER IN NS-2 APPLIED TO IMPULSE-RADIO ULTRA-WIDE BAND

2.1 Implementation Issues in Wireless Physical Layer of the NS-2

We describe and discuss what we believe are the current design issues and features missing in the current implementation of the ns-2 wireless physical layer.

2.1.1 Dependence on the 802.11 PHY and MAC

Today, there is a strong interdependence between the wireless physical layer implementation and the MAC layer implementation of 802.11 in ns-2. It is very hard to actually extend the current wireless physical layer implementation without changing parts of the 802.11 implementation. Indeed, in the current wireless physical layer of ns-2, when a packet starts to be received, it is directly delivered to the MAC layer. Packet reception actually occurs in the MAC layer rather than in the physical layer.

Furthermore, there are also various dependencies on the rest of the codebase in ns-2. Consequently, adding a new wireless physical layer requires many non-trivial changes in several sections of the code.

2.1.2 Lack of Multiple Sub-channels

All modern physical layers offer the possibility to share their available spectrum into several sub-channels. Sub-channels can appear in different ways, for instance:

- By having multiple transmission frequencies. A typical example is 802.11 where there are fourteen available transmission frequencies.
- With spread-spectrum physical layers, sub-channels appear naturally. Using either *direct-sequence* modulation as in direct-sequence CDMA (DS-CDMA), *frequency hopping* as in frequency-hopping CDMA (FH-CDMA) or *time-hopping* as with IR-UWB physical layers with time-hopping. A hybrid combination of these techniques is also possible. The reader is referred to [22] for an excellent explanation and details about direct-sequence and frequency hopping and to [22, 26] for time-hopping.

Unfortunately, there is no support currently in ns-2 for such a feature. Hence, it is not possible to simulate a scenario

[3]An alternate physical layer, based on IR-UWB, for the IEEE 802.15.4 standard. The IEEE 802.15.4 standard is also know as Zigbee.

[4]Note that these lookup tables need only be computed once for a given combination of modulation, coding, multi-user statistic and receiver implementation.

of 802.11 stations in the infrastructure mode with several access points using different frequencies or an IR-UWB network with the current implementation of ns-2.

2.1.3 Simplistic Model of Packet Detection and Timing Acquisition

Packet detection models the detection of a packet on the wireless channel, which in ns-2 is performed using a simple threshold for received signal strength. For example, with IR-UWB physical layers, this operation necessitates active decoding of the received signal. It is typically much more error-prone than packet detection in narrowband radios.

Packet detection schemes are traditionally characterized by parameters such as probability of missed detection (the probability that a receiver misses a packet) and probability of false alarm (the probability that a receiver believes it has detected a packet when there is actually no transmission).

After having detected that there is a packet on the channel, timing acquisition consists in detecting exactly when the payload of the packet begins. This is important for a proper demodulation and decoding of the payload. Any mistiming will lead to a degraded performance of the demodulation and decoding of the payload.

2.1.4 Absence of Error Model

The current model for packet reception in ns-2 assumes that a packet is properly received if the received power is higher than a given threshold and no single interferer is strong enough to cause a collision. No bit errors can occur during the packet transmission.

2.1.5 No Model of Cumulative Interference

The current model for packet reception does also not take into account the effect of interference from concurrent transmissions in the network. It only considers the received power from the source of the packet[5]. Obviously, if there are many ongoing transmissions from other stations in the network, the probability that the packet is correctly received is lower than if there is no interference.

2.2 Key Features of our Modified Wireless Physical Layer Architecture for NS-2

In this section, we describe the key features of our modified physical layer architecture for ns-2. We make a few important assumptions: (1) the physical layer cannot transmit and receive a packet at the same time; (2) it can receive only one packet at a time (no multi-user reception) (3) it can listen on more than one sub-channel.

2.2.1 Complete packet reception at the physical layer

Before being passed to the MAC layer, the packet is first completely received at the physical layer. At the end of the reception of the packet, the PER is calculated. Only then is the packet delivered to the upper layer.

2.2.2 Multiple transmission sub-channels

It is possible to specify a particular transmission sub-channel for each packet to be transmitted. Conversely, the physical layer can listen on more than one sub-channel. Typically, the physical layer would listen to a broadcast sub-channel and a receive sub-channel.

In our case, we have implemented support for multiple transmission sub-channels by adding a specific header to each packet. This header contains the index of the sub-channel used for transmitting the packet. The field of the header corresponding to the particular transmission sub-channel is set at the physical layer before the packet is passed to the wireless channel in order to be distributed to the stations in the network. When the reception of a packet begins, the physical layer can read the field corresponding to the sub-channel to check whether it corresponds to the one the physical layer is currently listening to.

The number of sub-channels and their relative orthogonality (i.e. whether there is interference between transmissions on different sub-channels) depends on the particular implementation.

2.2.3 Packet Detection and Timing Acquisition

In order to add support for packet detection and timing synchronization, we implemented an additional SYNC state to the physical layer. Hence the states of the physical layer are

- IDLE: the physical layer listens to the medium.
- SYNC: the physical layer believes it has detected a packet on the wireless medium and attempts to synchronize with the beginning of this packet.
- RECV: the physical layer receives the packet. It assumes that the physical layer has correctly detected that there is packet and is synchronized with its beginning.
- SEND: the physical layer transmits a packet.

Furthermore, there is a *detection and acquisition preamble* assumed for each transmitted packet. The length of this preamble is t_{pr} seconds. When the physical layer begins to receive a packet and it is in the IDLE state, it enters the SYNC state. The physical layer then sets the `end_of_timing_acquisition` timer to expire t_{pr} seconds later and adds the packet to the *synchronization list*. The synchronization list keeps a list of all the candidate packets for detection and acquisition while the physical layer is in the SYNC state. If the packet is not transmitted on a sub-channel that the receiver is currently listening to, the packet can still be added to the synchronization list, but with a very small probability.

If the physical layer is not in the IDLE state but already in the SYNC state, it does not prevent a packet from being potentially received. Instead, it directly adds this packet to the synchronization list.

Finally, when the `end_of_timing_acquisition` timer expires, one particular packet is selected from the synchronization

[5] Except for the capture effect.

list. How this packet is selected depends on the particular implementation. The remaining packets are not received but considered as interference. They are added to the *interference list* (see Section 2.2.4).

2.2.4 Cumulative Interference

Cumulative interference is considered for the whole duration of the transmission of the packet. The cumulative interference is the sum of the interferences created by simultaneous transmissions of *interfering packets* from other stations in the network. Interfering packets are packets obtained from the wireless channel by the physical layer that cannot be received, for instance, when the physical layer is transmitting a packet, or when the physical layer is already receiving a packet. Note that these interfering transmissions might occur on the same channel as the one used for the reception of the packet, or on another channel.

In order to implement this feature, we use an interference list at the physical layer in order to keep track of interfering transmissions. The following information about interfering packets is put in the interference list:

- The time corresponding to the beginning of the transmission of the interfering packet.
- The time corresponding to the end of the transmission of the interfering packet.
- The power at which the interfering packet is received.
- The sub-channel on which the interfering packet is transmitted.

Then, whenever a packet is completely received by the physical layer, the cumulative interference during the transmission of this packet is calculated for use in the error model (see Section 2.2.5).

2.2.5 Implementation of an Error Model

At the end the reception of packet, the following three steps are performed:

1. The cumulative interference during the transmission of the packet is calculated.
2. The cumulative interference is used to compute the average SINR during the transmission of the packet.
3. The average SINR is used to compute the PER of the packet. The PER is then used as the parameter of a binary random variable used to decide whether the packet is properly received or not.

How the PER is calculated depends on the particular physical layer implemented.

2.2.6 Flexibility when Computing the Channel and Packet Statistics

Our architecture is designed in a way that easily allows for the replacement of the particular physical layer implementation. The general architecture can be kept, but the following items must be modified; packet detection and timing synchronization, the calculation of the cumulative interference, the modelling of interference from other sub-channels, and the calculation of the PER.

2.3 An Impulse-Radio Ultra-Wide Band Physical Layer for NS-2

In this section, we detail the implementation-specific aspects of the previous section in the case of our IR-UWB physical layer.

2.3.1 Physical Layer Characteristics, Modulation and Channel Coding

Our physical layer implementation currently models an IR-UWB radio with time-hopping [26, 23] and a variable rate channel code. With IR-UWB, a sub-channel corresponds to the time-hopping sequence used by a transmitter.

The modulation is binary phase shift keying (BPSK). The channel codes are rate-compatible punctured convolutional codes (RCPC codes) [12, 9]. We use the codes proposed in [9], which offer a set of thirty possible rates.

2.3.2 Packet Detection and Timing Acquisition Model for IR-UWB

As explained in the introduction, packet detection and timing acquisition in IR-UWB networks is more challenging than in narrowband networks. However, it has interesting properties; if several packets are sent from different sources to the same destination at roughly the same time, all the packets sent with a time-hopping sequence that the receiver is listening to will trigger packet detection and timing acquisition at the receiver concurrently. In this case, with a very high probability, one packet will be acquired [7].

We implement the packet detection and timing acquisition mechanism described in [7]. Our implementation is described in detail in [19]. In the following, we explain how a particular packet is selected from the synchronization list and how packets are inserted in the synchronization list depending on the sub-channel (i.e. time-hopping sequence for IR-UWB) the packet was transmitted on.

At the end of the synchronization phase, a packet needs to be selected from the synchronization list. In our case, the packet in the list is chosen randomly (with a uniform distribution). This packet is further received by the physical layer with a probability $1 - P_{\mathrm{MD}}$, where P_{MD} is the probability of missed detection. The value of P_{MD} depends on the current level of interference, i.e. on the number of packets sent with a time-hopping sequence other than the ones the receiver is listening to.

How packets are inserted into the list depends on the sub-channels the potential receiver is currently listening to. We add to the synchronization list all the packets that are sent on the same sub-channels that the receiver is currently listening to. For packets sent on the sub-channels that the receiver is not listening to, we add them to the list with a probability Θ that depends on the particular algorithm used for packet detection and timing acquisition (see [19] for a description of how Θ is calculated).

2.3.3 Cumulative Interference and SINR

For a given packet being received from station i and concurrent transmissions of packets from stations $k \neq i$, the following factors are taken into account when computing the cumulative interference:

- The received power $P_{rx}^{(k)}$ from the kth station.
- The time $T_{overlap}^{(k)}$ during which the transmission of the packet from station i overlaps with the transmission of the packet from station k.
- A parameter Γ that takes into account the average orthogonality with respect to the transmissions using different time-hopping sequences and a parameter γ that takes into account the orthogonality between transmissions using the same time-hopping sequence. The parameters Γ and γ are computed following the expressions in [6, Equ. 12].

Hence, the cumulative interference I_c is

$$I_c = \Gamma \sum_{k \neq i} T_{overlap}^{(k)} P_{rx}^{(k)} + \gamma \sum_{l \neq i} T_{overlap}^{(l)} P_{rx}^{(l)} \qquad (1)$$

Then, with $P_{rx}^{(i)}$ the received power from the station i and N_{th} the thermal noise, the SINR is $\frac{P_{rx}^{(i)}}{I_c + N_{th}}$.

2.3.4 PER Calculation

In its current implementation, the PER is calculated as follows. For a given SINR and a given channel code rate, a BER value is obtained using a lookup table and linear interpolation. The PER is then calculated as $PER = 1-(1-BER)^L$ where L is the length of the payload. The lookup tables were computed offline with extensive Matlab simulations. There is one lookup table for each possible rate of the codes in [9]. Note that these lookup tables need only be computed once for a given combination of modulation, coding, multi-user statistic and receiver implementation.

3. END-TO-END PATH OF A PACKET THROUGH THE MAC AND THE PHYSICAL LAYER

This section describes the path of a packet through our physical layer implementation.

- The MAC layer has a packet ready to be sent to the physical layer. The MAC layer checks whether the physical layer is idle or not. If it is idle, the MAC layer sends the packet to the physical layer.
- The physical layer receives the packet from the MAC layer. First, if the physical layer not idle, the packet is dropped. Else, the state of the PHY layer is set to SEND[6]. Then, the physical layer sets the transmission rate (i.e. the proper modulation and coding), computes the transmission time and sets the particular time-hopping sequence.
- The physical layer places the packet on the channel. The channel delivers the packet to the physical layer of other nodes.

As several nodes might receive the packet, the following steps might be executed by several nodes.

- First, the power of the packet received from the channel is computed. The computation is based on the propagation model in [10]. Then, a set of tests are applied on the packet to check the following conditions:
  - If the physical layer is not busy transmitting (SEND state) or receiving a packet (RECV state).
  - If the receiver is listening to with the same time-hopping sequence as the time-hopping sequence used for transmitting this packet.

  If any of these tests fail, then the packet is an interfering packet and is put in the interference list. If, on the contrary, the packet satisfies these tests, then the packet detection and timing acquisition phase can start. Remember that if the physical layer is in the SYNC state, this does not prevent the packet from being received. The packet is added to the synchronization list.
- The physical layer of the receiving node performs packet detection. In its current form, the implementation consists in testing whether the received power of the packet is sufficiently high to trigger the packet detection and timing acquisition part. If so, then the packet detection is considered successful. The state of the physical layer is set to SYNC.
- The physical layer performs timing acquisition. This consists in adding the packet to the synchronization list. Note that the first packet that triggers the SYNC state also starts the timer scheduled to expire after t_{pr} seconds. When the timer expires, there will be at most one packet from the synchronization list for which the timing acquisition is successful. The receiver will have "locked" itself on this particular received packet and can proceed with the decoding of this packet. All the other packets from the synchronization list are added to the interference list and the synchronization list is emptied. The state of the PHY layer is set to RECV.
- The physical layer decodes the header and payload of the packet.
- When the packet reception is over, the PER is computed as explained in Sections 2.3.3 and 2.3.4. Finally, the PER is used to decide whether the packet is properly received or not and whether the physical layer delivers the packet to the MAC layer.

4. SIMULATIONS

We present several simulation results that show some of the features of our implementation. The parameters of our physical layer implementation correspond to a typical 802.15.4a physical layer with a bitrate of 1 Mbit/s. For the channel code, we use three different rates; code rate $\frac{8}{11}$, $\frac{1}{2}$ and $\frac{1}{3}$

[6] A timer set to expire at the end of the packet transmission sets the PHY layer state back to IDLE.

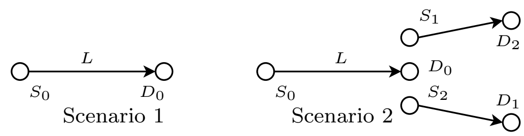


Figure 1: Scenarios used for the simulations. Scenario 1: a single source destination pair S_0 to D_0. Scenario 2: the same source-destination pair S_0 to D_0 but with two interfering sources S_1 and S_2 located at 1 meter from D_0. S_1 and S_2 transmit to D_1 and D_2 respectively which are 10 meters away.

corresponding to code index 2, 7 and 15 in [9]. For packet detection and timing acquisition, values for P_{MD} and Θ (see Section 2.3.2) are derived from [7] according to [19]. The MAC layer protocol used is DCC-MAC [20]. The transport protocol is UDP. The UDP agent in ns-2 is used with a maximum segment size of 1000 bytes. A constant bit rate (CBR) traffic generator with a packet size of 1000 bytes and random interpacket departure is attached to the UDP agent. The propagation model used is [10] in the line-of-sight case without the random component. Hence, path loss is a deterministic function of the distance. Note that IR-UWB networks have a very low transmit power. Hence the transmission range is of the order of a few tens of meters.

Our performance metric is the saturation throughput; this throughput is computed when sources always have a packet available to transmit and queuing at the sources is ignored. Each simulation was run ten times for a duration of 300 seconds. We calculated the 95% confidence intervals for the median for each set of runs.

We use two scenarios (see Figure 1). The first is a single source-destination pair where we vary the link distance L. The second scenario is again a source-destination pair with a variable link distance L, but with two sources located one meter from the receiver and transmitting to their respective destination ten meters away. This is a typical near-far scenario. Note that in both cases, we only look at the performance of the link S_0 to D_0. With the DCC-MAC protocol, sources transmitting to a given destination use a so-called *private* time-hopping sequence specific to the destination. Hence with the second scenario, S_0 and S_1 do not use the same time-hopping sequence. There are concurrent transmissions occurring on different sub-channels.

In Figure 2, we use the first scenario to illustrate the effect of the error model. We look at the saturation throughput as a function of the link distance for three different channel code rates. As the link distance increases, the received power and SINR at the destination decrease. This gradually increases the average PER, which leads to the smooth degradation of throughput.

For Figures 3(a) and 3(b), we use the second scenario to illustrate the effect of cumulative interference. We look again at the saturation throughput of the original source-destination link S_0 to D_0 of the first scenario, but this time, there is interference created by the two other sources. As can be clearly seen, the cumulative interference induces a net throughput reduction. Indeed, for a given link distance, the

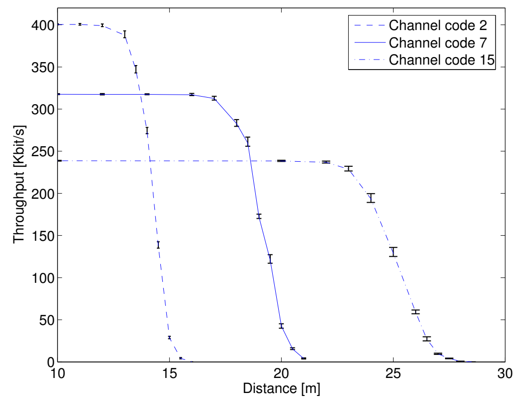


Figure 2: Saturation throughput of the link S_0 to D_0 versus the link distance L for three different channel code rates. The topology is scenario 1. Due to the error model, the throughput smoothly decreases with the distance.

cumulative interference reduces the SINR and consequently the PER is higher as seen in Figure 3(b).

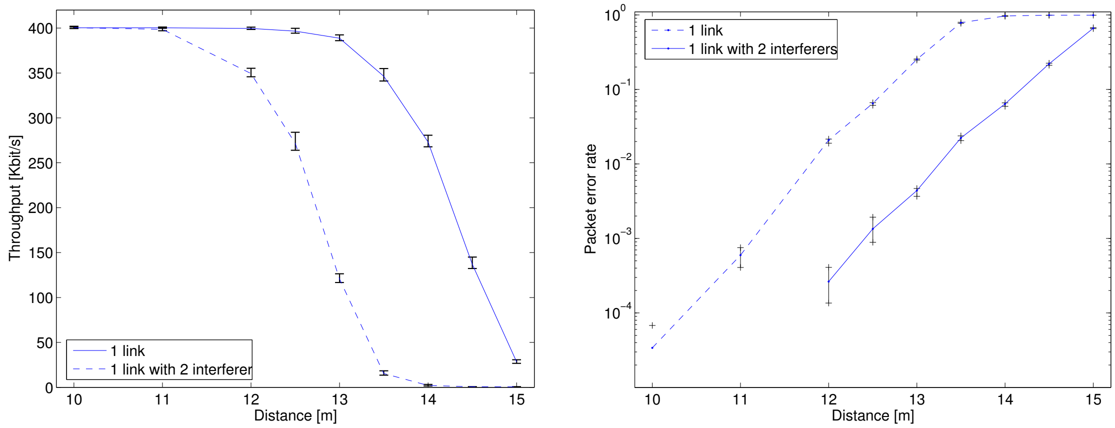
Finally, in Figure 4, we again use the second scenario, but this time to observe the impact of the packet detection and timing synchronization model. We compare the throughput of the link S_0 to D_0 versus the link distance for two cases. One where sources use the time-hopping sequence of the destination (private time-hopping sequences) and one where all sources are forced to use the same unique time-hopping sequence. In other words, one where there are sub-channels and one where all nodes share the same sub-channel. In the case of the single time-hopping sequence, the destination of the link of interest acquires many packets from the interferers, which greatly reduces the throughput. The slightly better throughput obtained for link distance 13.5 to 14.5 is explained from the fact that with a single time-hopping sequence, the interfering sources also receive packets from the source and are prevented from sending. As such, there are a few packets that are transmitted with less interference than in the case where sources use the time-hopping sequence of the destination. By using a code with a slightly lower rate but a better protection against interference, this difference disappears.

More complicated scenarios, such as a line of nodes with UDP or TCP, or random topologies, can be found in [20, 19].

5. RELATED WORK

To the best of our knowledge, there is currently no other model of an IR-UWB physical layer available for ns-2 or an other network simulator.

Networks simulators (see the excellent list of references in [17]) such as GloMoSim/Qualnet, Jist/SWANS, OMNET++, OPNET, yans [17] or GTNetS[24] allow for the use and implementation of an error model at the physical layer. However, none of them appears to implement sub-channels or to finely model an explicit packet detection and timing synchronization phase.



(a) Saturation throughput versus link distance with channel code rate 8/11.

(b) Packet error rate versus link distance with channel code rate 8/11.

Figure 3: Saturation throughput and packet error rate of the link S_0 to D_0 versus the link distance L with channel code rate 8/11. We compare scenario 1 (plain curve) with scenario 2 (dashed curve). Cumulative interference clearly degrades the throughput and increases the PER.

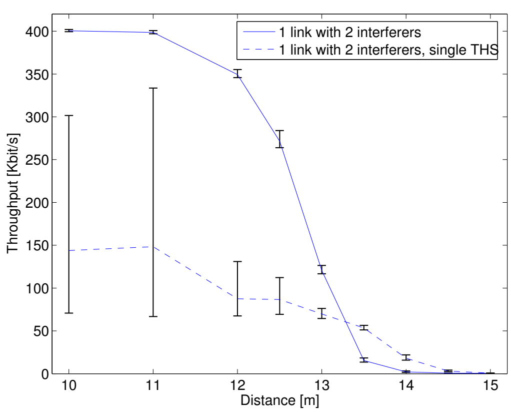


Figure 4: Saturation throughput of the link S_0 to D_0 versus the link distance L with channel code rate 8/11. The topology is scenario 2. Using a single time-hopping sequence (dashed curve) in the network decreases the throughput compared to private time-hopping sequences (plain curve). Indeed, in this case, the nodes can acquire packets not sent to them.

Regarding ns-2 in particular, the "Contributed Code" section of the ns-2 website[7] lists several extensions and modifications of ns-2. In the case of 802.11, [2] implements an error model based on a signal to noise ratio calculation with cumulative interference.

Still in the case of 802.11, [14, 5] both implemented an error model based on signal to noise ratio computation but did not take the cumulative interference into account. An interesting and more recent approach is [4] where the cumulative interference is taken into account; they do not use an error model, but rather declare a successful reception if the SINR is higher than a given threshold. Compared to our work, the previous approaches are unfortunately specific to 802.11. Furthermore, there is no implementation of sub-channels or a precise model of the packet detection and timing synchronization state. A very promising work in the case of 802.11 is [21]; they propose a model to take into account transmissions on multiple overlapping frequency bands.

It is worth noting the large body of work that addresses the impact of radio channel and propagation models on wireless network simulations. The reader can refer to [25, 16] and the references therein. Finally, [13] addresses the important issue of validation. Validation is currently very difficult in our case due to the lack of standard hardware. This situation should rapidly evolve with the successful completion of the IEEE 802.15.4a standard.

6. CONCLUSION AND FUTURE WORK

We have presented an architecture for wireless simulation in a packet based network simulator. We have used this architecture to implement an impulse-radio ultra-wide band physical layer in ns-2. Our architecture attempts to properly model the characteristics of modern physical layers: cu-

[7] `http://nsnam.isi.edu/nsnam/index.php/Contributed_Code`, October 2006

mulative interference and the calculation of a packet error rate, packet detection and timing synchronization, and the possibility to have multiple sub-channels.

Future work will integrate a better BER and PER calculation model for IR-UWB (such as [18] for instance). An important effort is also necessary to validate our physical layer model with actual hardware. With the emergence of the IEEE 802.15.4a standard, we will be able to adapt our model to this standard and refine the implementation. The calculation of the PER, the effect of cumulative interference, and the packet detection and timing acquisition phase are elements that need further enhancement and validation.

7. REFERENCES

[1] ns Network Simulator. `http://www.isi.edu/nsnam/ns/`.

[2] New 802.11 support for ns-2. `http://spoutnik.inria.fr/ns-2-80211/`, 2006.

[3] UWB research at EPFL-IC. `http://icawww1.epfl.ch/uwb/`, 2007.

[4] Q. Chen, D. Jiang, V. Taliwal, and L. Delgrossi. Ieee 802.11 based vehicular communication simulation design for ns-2. In *VANET '06*, pages 50–56, 2006.

[5] J. M. Dricot and P. De Doncker. High-accuracy physical layer model for wireless network simulations in ns-2. In *Wireless Ad-Hoc Networks, 2004 International Workshop on*, pages 249–253, 2004.

[6] G. Durisi and G. Romano. On the validity of gaussian approximation to characterize the multiuser capacity of UWB TH PPM. In *IEEE UWBST*, pages 157–161, 2002.

[7] A. El Fawal and J.-Y. Le Boudec. A robust signal detection method for ultra wide band (UWB) networks with uncontrolled interference. *IEEE Transactions on Microwave Theory and Techniques*, 2006, to appear.

[8] A. El Fawal, J.-Y. Le Boudec, R. Merz, B. Radunovic, J. Widmer, and G. M. Maggio. Tradeoff analysis of PHY-aware MAC in low-rate, low-power UWB networks. *IEEE Communications Magazine*, 43(12):147–155, December 2005.

[9] P. Frenger, P. Orten, T. Ottosson, and A. Svensson. Rate-compatible convolutional codes for multirate DS-CDMA systems. *IEEE Transactions on Communications*, 47(6):828–836, June 1999.

[10] S. Ghassemzadeh, R. Jana, C. Rice, W. Turin, and V. Tarokh. Measurement and modeling of an ultra-wide bandwidth indoor channel. *IEEE Transactions on Communications*, 52(10):1786–1796, October 2004.

[11] G. Giancola, L. De Nardis, and M.-G. Di Benedetto. Multi user interference in power-unbalanced ultra wide band systems: analysis and verification. In *IEEE UWBST*, pages 325–329, 2003.

[12] J. Hagenauer. Rate-compatible punctured convolutional codes (RCPC codes) and their applications. *IEEE Transactions on Communications*, 36(4):389–400, April 1988.

[13] J. Heidemann, K. Mills, and S. Kumar. Expanding confidence in network simulations. *Network, IEEE*, 15(5):58–63, 2001.

[14] G. Holland, N. Vaidya, and P. Bahl. A rate-adaptive mac protocol for multi-hop wireless networks. In *MobiCom '01*, pages 236–251, 2001.

[15] B. Hu and N. Beaulieu. Accurate evaluation of multiple-access performance in TH-PPM and TH-BPSK UWB systems. *IEEE Transactions on Communications*, 52(10):1758–1766, October 2004.

[16] D. Kotz, C. Newport, R. S. Gray, J. Liu, Y. Yuan, and C. Elliott. Experimental evaluation of wireless simulation assumptions. In *MSWiM '04*, pages 78–82, 2004.

[17] M. Lacage and T. R. Henderson. Yet another network simulator. In *WNS2, The workshop on ns-2: the IP network simulator*, 2006.

[18] R. Merz and J.-Y. Le Boudec. Conditional bit error rate for an impulse radio UWB channel with interfering users. In *IEEE International Conference on Ultrawideband*, September 2005.

[19] R. Merz, J.-Y. Le Boudec, and S. Vijayakumaran. Effect on network performance of common versus private acquisition sequences for impulse radio uwb networks. In *IEEE International Conference on Ultrawideband*, September 2006.

[20] R. Merz, J. Widmer, J.-Y. Le Boudec, and B. Radunovic. A joint PHY/MAC architecture for low-radiated power TH-UWB wireless ad-hoc networks. *WCMC Journal, Special Issue on Ultrawideband (UWB) Communications*, 5(5):567–580, August 2005.

[21] A. Mishra, V. Shrivastava, S. Banerjee, and W. Arbaugh. Partially overlapped channels not considered harmful. In *SIGMETRICS '06*, pages 63–74, 2006.

[22] R. Pickholtz, D. Schilling, and L. Milstein. Theory of spread-spectrum communications–a tutorial. *IEEE Transactions on Communications*, 30(5):855–884, 1982.

[23] R. C. Qiu, H. Liu, and X. Shen. Ultra-wideband for multiple access communications. *IEEE Communications Magazine*, 43(2):80–87, 2005.

[24] G. F. Riley. The georgia tech network simulator. In *MoMeTools '03*, pages 5–12, 2003.

[25] M. Takai, J. Martin, and R. Bagrodia. Effects of wireless physical layer modeling in mobile ad hoc networks. In *MobiHoc '01*, pages 87–94, 2001.

[26] M. Z. Win and R. A. Scholtz. Impulse radio: how it works. *IEEE Communications Letters*, 2(2):36–38, 1998.